AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions and listings of claims in the application.

LISTING OF CLAIMS

1. (currently amended) A method for controlling a value of a RAM variable inside an executable program, comprising:

presenting a software program in executable form and having a plurality of machine instructions of a finite quantity of fixed lengths;

selecting a parameter of interest that is defined in random access memory associated with the software program;

evaluating each machine instruction in the executable form of the software program to identify machine instructions that contain address information for the parameter;

~~identifying at least one machine instruction that accesses a variable defined in random access memory associated with the software program;~~

replacing the identified machine instruction~~s~~ in the executable form of the software program with a branch instruction that references an address outside an address space of the software program;

defining a set of relocated instructions at the address referenced by the branch instruction, wherein the set of relocated instructions function to change a value of the variable; and executing the executable form of the software program having the branch instruction.

2. cancelled.

3. cancelled.

4. cancelled.

5. (currently amended) The method of Claim 1 wherein the step of evaluating each machine instruction further comprises determining whether a given machine instruction loads an address for the parameter into a machine register; and evaluating each instruction nested in the given machine instruction to further identify machine instructions that contain address information for the parameter when the given machine instruction loads an address for the parameter; ~~identifying at least one machine instruction further comprises determining an address for the at least one machine instruction within the software program~~.

6. (currently amended) The method of claim 1 further comprises replacing machine instructions identified as a load instruction with a branch instruction ~~Claim 5 wherein the step of determining location information further comprises identifying an address for the at least one machine instruction using the image of the executable containing the machine instructions that comprise the executable~~.

7. (currently amended) The method of Claim 6 wherein the step of replacing ~~the at least one~~ machine instruction further comprises inserting the branch ~~replacement~~ instruction into a program memory image of the software program at said address.

8. (previously presented) The method of Claim 1 wherein said branch instruction references a set of relocation instruction residing at an unused address space of the software program.

9. cancelled.

10. (currently amended) A computer-implemented calibration system for modifying RAM variables of a software program in an executable form and having a plurality of machine instruction of a finite quantity of fixed length embedded in a microprocessor, comprising:

an instruction locator embodied as computer executable instructions on a computer readable medium and operating on a different processor than the microprocessor, the instruction locator that selects a parameter of interest in the software program that is defined in random access memory associated with the software program and evaluates each machine instruction in the executable form of the software program to identify machine instructions that contain address information for the parameter ~~adapted to receive an address for RAM variable within an software program and operable to identify at least one machine instruction in an executable form of the software program that accesses the RAM variable~~; and an instruction replacement component embodied as computer executable instructions on a computer readable medium and operating on the different processor than the microprocessor and in data communication with the instruction locator, the instruction replacement component adapted to receive a branch instruction for the at least one machine instruction and operable to replace the identified ~~at least one~~ machine instructions in the executable form of the software program with the branch instruction.

11. (currently amended) The computer-implemented calibration system of Claim 10 wherein the instruction locator evaluates each machine instruction to determine whether a given machine instruction loads an address for the parameter into a machine register; and evaluates each instruction nested in the given machine instruction to further identify machine instructions that contain address information for the parameter when the given machine instruction loads an address for the parameter ~~is operable to identify an address for the specified machine instruction using the image of the executable containing the machine instructions that comprise the executable~~.

12. (currently amended) The computer-implemented calibration system of Claim 11 wherein the instruction replacement component is operable to insert the replacement instruction into a program memory image of the software program at said address.

13. cancelled.

14. (currently amended) The computer-implemented calibration system of Claim 10 wherein the instruction replacement component is operable to generate a set of relocation instructions, such that the branch instruction passes processing control to the set of relocation instructions.

15. (currently amended) The computer-implemented calibration system of Claim 14 wherein the instruction replacement component is further operable to insert the set of relocation instructions in a memory space of the microprocessor that resides at an unused address space of the software program.

16. (currently amended) A method for controlling the value of a RAM variable inside an executable program, comprising:

presenting a software program in executable form and having a plurality of machine instructions of a finite quantity of fixed lengths;

selecting a parameter of interest that is defined in random access memory associated with the software program;

evaluating each machine instruction in the executable form of the software program to identify machine instructions that contain address information for the parameter;

~~identifying at least two machine instructions that accesses a variable defined in random access memory associated with the software program;~~

replacing each of the identified machine instruction in the executable form of the software program with a branch instruction when the identified machine instruction is a load instruction or a store instruction, where each branch instruction references a different address outside an address space of the software program;

evaluating each of the identified machine instructions by searching for additional machine instructions that contain address information for the parameter and are referenced by the identified machine instructions when the identified machine instruction is not a load instruction nor a store instruction;

defining a set of relocated instructions at each address referenced by the branch instructions, wherein each set of relocated instructions accesses the variable in random access memory and performs an operation to change a value of the variable in a different manner; and

executing the executable form of the software program having the branch instruction.

17. – 19. cancelled.